

Sistemas Operacionais

Sistema de arquivos 2

Cynthia da Silva Barbosa

© 2018 por Editora e Distribuidora Educacional S.A.

Todos os direitos reservados. Nenhuma parte desta publicação poderá ser reproduzida ou transmitida de qualquer modo ou por qualquer outro meio, eletrônico ou mecânico, incluindo fotocópia, gravação ou qualquer outro tipo de sistema de armazenamento e transmissão de informação, sem prévia autorização, por escrito, da Editora e Distribuídora Educacional S.A.

Sumário

Unidade 3 Sistema de arquivos 2	5
Seção 3.1 - Arquivos e sistemas de arquivos	7
Seção 3.2 - Sistemas de diretórios	21
Seção 3.3 - Implementação e segurança de sistemas de arquivos	36

Sistema de arquivos 2

Convite ao estudo

Caro aluno, seja bem-vindo!

Nesta unidade, veremos como os arquivos são usados pelo usuário, suas características, nomeação e estrutura, além de apresentarmos os tipos suportados pelos sistemas operacionais. Estudaremos também como é realizado o acesso, os atributos dos arquivos e as operações realizadas nos mesmo, como se dá a hierarquia de diretórios e como são realizados o acesso e as operações. Por fim, abordaremos a implementação de arquivos e diretórios, o gerenciamento de espaço em disco e como se dá a segurança e a confiabilidade do sistema de arquivos. Os arquivos armazenam informações em disco e, posteriormente, permitem que essas informações sejam recuperadas. Um sistema de arquivos é um conjunto organizado de arquivos e diretórios. No Windows e no Linux, existe um diretório raiz em que os diretórios e arquivos ficam vinculados. Por exemplo, dentro do diretório "Computador" do Windows existe uma hierarquia de diretórios vinculados a ele, tais como os diretórios C:, D:, E:, dentre outros que possam existir. No Linux, o diretório raiz (/) contém os seguintes diretórios: /etc (configurações do sistema), /home (arquivos pessoais), /boot (arquivos de inicialização do sistema), entre outros. Nesse contexto, você é um engenheiro da computação que foi contratado pela empresa XYZ Ltda., que atua no ramo de seguros automotivos e residenciais, com filiais em todo o país, para organizar os dados e implementar o seu sistema de arquivos. Para isso, é necessário que você tenha domínio de como os arquivos são organizados logicamente, como o sistema operacional facilita a interação do usuário com os arquivos, os tipos de arquivos permitidos, as operações que podem ser realizadas com arquivos e as estruturas de diretórios. A empresa XYZ Ltda. possui os sistemas operacionais Linux e Windows e movimenta um grande volume de dados. Você deve fazer a escolha certa, de acordo com o uso dos sistemas da empresa, para que futuramente não ocorram lentidão ou outros problemas. Para iniciar a operação, será necessário realizar as seguintes etapas:

- Nomear os arquivos, definir a sua estrutura, apresentar os tipos de arquivos suportados, o acesso aos arquivos deve ser de forma aleatória, definir os atributos e operações a serem realizadas nos arquivos;
- II. Definir os diretórios para controle dos arquivos, especificar os nomes de caminhos e as operações com os diretórios; e
- III. Implementar e garantir a segurança do sistema de arquivos.

Como poderão ser definidos os nomes e a estrutura de arquivos para a empresa XYZ Ltda.? Como poderão ser definidas a hierarquia de diretórios e as operações a que cada usuário terá acesso? Como será implementada a segurança de arquivos? Ao final, você deverá entregar um anteprojeto para a avaliação dos gestores da empresa, apresentando todos os dados obtidos no trabalho realizado. Após o término desta unidade, você terá condições de entender a organização de arquivos e diretórios nos sistemas operacionais, as operações que podem ser realizadas e como podem ser implementados os sistemas de arquivos e diretórios.

Vamos juntos conhecer mais sobre os arquivos e suas propriedades?

Seção 3.1

Arquivos e sistemas de arquivos

Diálogo aberto

O sistema de arquivos é a maneira como as informações são armazenadas nos dispositivos (disco rígido, pen drives, dentre outros) e os arquivos são a maneira que o sistema operacional permite a leitura e gravação dos dados. Um arquivo é composto por um nome e por atributos (tamanho do arquivo, código de proteção de acesso, quem é o criador do arquivo, data de criação do arquivo) que são gerenciados pelo sistema operacional. Por exemplo, o arquivo TESTE.TXT, é um arquivo do tipo texto, com o nome TESTE e possui a extensão de .TXT (arquivo de texto). Nesta seção você conhecerá como os arquivos são usados pelo usuário, suas características, como se dá a nomeação e a estrutura de arquivos, os tipos de arquivos suportados pelos sistemas operacionais, como é realizado o acesso aos arquivos, os atributos dos arquivos e as operações realizadas em arquivos. Relembrando nosso contexto, você é um engenheiro da computação que foi contratado pela empresa XYZ Ltda. do ramo de seguros automotivos.

A empresa relata que tem perdido arquivos importantes, dado que a forma de armazenamento dos mesmos é arcaica, sendo dispostos em caixas em um depósito. Além do risco de degradação por umidade ou outras intempéries, tem-se a dificuldade da busca de documentos específicos, fatores estes que prejudicam as atividades da mesma. Os arquivos de cadastros de clientes de todas as filiais da empresa XYZ Ltda. estão armazenados em caixas, num depósito na matriz, localizada em São Paulo. As caixas estão separadas por cidades em que a empresa possui filiais e por nomes, como pré-vendas, vendas e boletos. Por exemplo: Belo Horizonte – Vendas, Rio de Janeiro – Boletos, Salvador – Prévendas. Uma parte desse cadastro foi armazenada em CDs-ROM e outra parte em HD.

Desta forma, você foi encarregado de organizar e implementar um sistema de arquivos. Portanto, surgem os seguintes questionamentos: quais levantamentos devem ser feitos antes de iniciar os trabalhos com os arquivos da empresa XYZ Ltda.? Qual sistema de arquivos será implementado nos sistemas operacionais Linux e Windows? Como serão definidos os nomes, atributos, o acesso e as operações dos arquivos da empresa XYZ Ltda.? Quais são operações a serem realizadas com os arquivos? Para exemplificar os conceitos e justificar a implementação do sistema aos executivos da empresa, os gestores solicitaram que utilize os documentos referentes ao "Levantamento de Vendas Mensais", que contêm informações de todas as vendas realizadas no mês, por setor. Lembre-se que estas informações o auxiliarão na elaboração do anteprojeto de um sistema de proteção de arquivos.

Para que você consiga responder a esses e outros questionamentos sobre os arquivos, nesta seção vamos conhecer mais sobre eles e os conteúdos pertinentes a este tema.

Bons estudos!

Não pode faltar

As aplicações armazenam e recuperam informações durante seu processamento, e um processo deve ser capaz de ler e gravar grandes volumes de informações em disco rígido, pen drives, entre outros, além de dividir as informações com outros processos. Segundo Tanenbaum (2003), existem três requisitos essenciais para o armazenamento da informação a longo prazo:

- Deve ser possível armazenar um grande volume de informações (a exemplo dos sistemas bancários, companhias aéreas, entre outros).
- A informação deve sobreviver ao término do processo que a usa (armazenamento em banco de dados);
- Múltiplos processos têm que ser capazes de acessar a informação concorrentemente (a informação deve ser independente de qualquer processo).

O sistema operacional organiza e estrutura essas informações por meio de arquivos.

De acordo com Tanenbaum (2003), um arquivo pode ser definido como um mecanismo de abstração, oferecendo meios de

armazenamento de dados e permitindo que esses dados sejam lidos posteriormente. Segundo Machado e Maia (2007), o responsável por gerenciar os arquivos no sistema operacional é o sistema de arquivos, que é a parte visível de um sistema operacional, uma vez que os usuários manipulam constantemente os arquivos. Assim, é importante que o sistema de arquivos possua uma interface amigável e de fácil implementação. Cada sistema operacional possui seu próprio sistema de arquivos.

Nomeação de arquivos

Quando um processo cria um arquivo, atribui-se um nome a tal arguivo, e quando o processo é encerrado, o arguivo continua existindo e outros processos tem acesso a ele, buscando-o pelo seu nome (TANENBAUM, 2003). As regras de nomeação de arquivos variam de acordo com o sistema operacional. Os sistemas operacionais atuais aceitam cadeias de caracteres com entre uma e oito letras como nomes válidos de arquivos. Caracteres especiais e nomes com tamanhos de até 255 caracteres são aceitos em alguns sistemas de arquivos, por exemplo o Unix. O sistema de arquivos do Unix faz distinção entre letras maiúsculas e minúsculas (case sensitive), já o sistema de arquivos do Windows não faz esta distinção. A identificação do arquivo é composta por duas partes, separadas por um ponto, sendo que a parte após o ponto é denominada extensão do arquivo e identifica o conteúdo do arquivo (MACHADO; MAIA, 2007). O Quadro 3.1 apresenta algumas extensões de arquivos.

Quadro 3.1 | Algumas extensões de arquivos

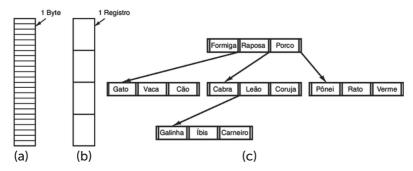
Extensão	Significado
Arquivo.txt	Arquivo de textos
Arquivo.zip	Arquivo Comprimido
Arquivo.jpg	lmagem codificada com o padrão JPEG
Arquivo.c	Programa fonte em C
Arquivo.bak	Arquivo de cópia de segurança
Arquivo.pdf	Arquivo no formato portátil de documentos
Arquivo.mp3	Música codificada de áudio MPEG

Fonte: adaptado de Tanenbaum (2003, p. 59).

Estrutura dos arquivos

Segundo Tanenbaum (2003), os arquivos podem ser estruturados de várias formas, sendo que as três mais comuns são: sequência estruturada de bytes, sequência de registro de comprimento fixo e árvore de registros. A Figura 3.1 apresenta a estrutura dos arquivos.

Figura 3.1 | Estruturas de arquivos



Fonte: Tanenbaum (2003, p. 288).

- Sequência estruturada de bytes: na estrutura da Figura 3.1 (a), o sistema operacional não sabe qual é o conteúdo do arquivo, e tudo o que ele vê são bytes. A sequência estruturada de bytes oferece flexibilidade, uma vez que os programas de usuário podem dar o nome que quiserem aos seus arquivos e inserir o conteúdo que desejarem.
- Sequência de registro de comprimento fixo: na estrutura da Figura 3.1 (b), um arquivo é uma sequência de registros de tamanho fixo, cada um com uma estrutura interna. O objetivo é que a operação de leitura retorne um registro e a operação de escrita sobreponha ou anexe um registro.
- Árvore de registros: na estrutura da Figura 3.1 (c), um arquivo é formado por uma árvore de registros, não necessariamente do mesmo tamanho, cada um contendo um campo-chave em uma posição fixa do registro. A árvore é ordenada pelo campo-chave, para que se busque mais rapidamente por uma chave específica. Novos registros podem ser inseridos no arquivo, e é o sistema operacional que decide onde colocá-los. Esse tipo de arquivo é amplamente aplicado em computadores de grande porte, usados ainda para alguns processamentos de dados comerciais.

Tipos de arquivos

Diferentes tipos de arquivos são suportados pelos sistemas operacionais, tais como arquivos regulares, diretórios, arquivos especiais de caracteres e especiais de blocos.

Tanembaum (2003) afirma que os arquivos regulares contêm informações do usuário, e são classificados em:

- Arquivos ASCII: são formados por linhas de texto. A grande vantagem dos arquivos ASCII é que eles podem ser mostrados e impressos como são e podem ser editados com qualquer editor de textos, além de facilitar a conexão entre a saída de um programa e a entrada de um outro.
- Arquivos binários: possuem alguma estrutura interna, conhecida pelos programas que os usam. Os sistemas operacionais devem reconhecer pelo menos um tipo de arquivo: seu próprio arquivo executável.

Os arquivos de diretórios mantêm a estrutura dos sistemas de arquivos. Já os arquivos especiais de caracteres são relacionados à entrada/saída e usados para modelar dispositivos de E/S, como terminais, impressoras e rede. Os arquivos especiais de blocos são usados para modelar discos.

Acesso aos arquivos

Os arquivos podem recuperar informações de diferentes formas, de acordo com sua organização. Segundo Machado e Maia (2007), os primeiros sistemas operacionais armazenavam os arquivos em fitas magnéticas e seu acesso era realizado de forma sequencial. No acesso sequencial, os arquivos são lidos sequencialmente, a partir do início do arquivo, sempre na ordem em que os registros foram gravados.

A Figura 3.2 apresenta um modelo de acesso sequencial. A leitura do arquivo era realizada avançando o ponteiro do arquivo para o próximo registro, e a gravação era realizada sempre no final do arquivo.

Figura 3.2 | Acesso sequencial
Início
Posição Corrente
Fim
Rebobinar
Leitura ou Gravação

Fonte: Galvin et al (2013, p. 431).

Com a criação dos discos magnéticos, surgiu o acesso direto, cuja leitura/gravação é feita na sua posição por meio do número do registro. Não existem restrições em relação à ordem em que os registros são lidos ou gravados, sendo sempre necessário especificar o número do registro. O acesso direto somente é possível quando os registros do arquivo são de tamanho fixo, sendo que o mesmo pode ser combinado com o acesso sequencial, sendo possível acessar qualquer registro de um arquivo e, assim, acessar os demais.

Um método mais sofisticado é o acesso indexado ou por chave. O arquivo deve possuir uma área de índice em que existam ponteiros para os diversos registros. A aplicação que desejar acessar o registro deverá especificar uma chave para que o sistema pesquise, na área de índice, o ponteiro correspondente, acessando o arquivo diretamente.

Atributos dos arquivos

Segundo Machado e Maia (2007), atributos são informações de controle de cada arquivo, que variam de acordo com o sistema de arquivos, mas o tamanho do arquivo, a proteção, a identificação do criador e a data de criação estão presentes em quase todos os sistemas. Alguns atributos especificados na criação de arquivos não podem ser modificados, como a organização e data/hora da criação. Outros atributos podem ser alterados pelo sistema operacional, como o tamanho e a data/hora do último backup realizado. Existem, ainda, atributos que podem ser modificados pelo usuário, como proteção do arquivo, tamanho máximo e senha de acesso.

Observe no Quadro 3.2 alguns atributos dos sistemas de arquivos:

Quadro 3.2 | Atributos e caraterísticas de arquivos

Atributo	Característica		
Proteção	Quem pode ter acesso ao arquivo e de que forma.		
Senha	Para ter acesso ao arquivo.		
Criador	ID do criador do arquivo.		
Proprietário	Atual proprietário.		
Tamanho do registro	Número de bytes em um registro.		
Tamanho Atual	Número de bytes no arquivo.		
Momento da Criação	Data/horário da criação do arquivo.		

Fonte: adaptado de Tanenbaum (2003, p. 291).



Vimos nesta seção que os atributos são informações de controle de cada arquivo e variam de acordo com o sistema de arquivos. Além dos atributos apresentados nesta seção, temos os seguintes atributos: "somente leitura", que indica que o arquivo não deve ser modificado ou deletado, e o atributo "oculto", usado quando um arquivo não deve ser visto por todos.

Pesquise mais

Para garantir a segurança dos seus arquivos na nuvem, é necessário seguir alguns passos importantes: criar uma senha forte, não utilizar redes WiFi públicas para acesso aos arquivos, reforçar a autenticação das contas, manter seus programas e aplicativos atualizados e não permitir o backup automático dos seus dados. Para saber mais sobre esse assunto, consulte o artigo:

WALTRICK, R. Saiba como proteger seus arquivos salvos na nuvem. **Gazeta do Povo**. 3 set. 2014. Disponível em: http://www.gazetadopovo.com.br/tecnologia/saiba-como-proteger-seus-arquivos-salvos-na-nuvem-ecz71ltnny743iyd4u3yeas7i>. Acesso em: 8 maio 2018.

Operações com arquivos

Segundo Tanenbaum (2003), sistemas de arquivos diferentes oferecem operações distintas para armazenar e recuperar informações. As principais operações com arquivos são:

- <u>Create:</u> cria um arquivo sem dados e define alguns atributos.
- <u>Delete</u>: deleta um arquivo quando ele não é mais necessário, liberando espaço de memória.
- Open: um processo deve abrir um arquivo antes de usá-lo, permitindo que o sistema busque e coloque na memória principal os atributos e a lista de endereços do disco, agilizando o acesso as informações.
- <u>Close:</u> fecha um arquivo e libera espaço na memória.
- Read: leitura dos dados do arquivo.
- <u>Write:</u> grava os dados no arquivo.

Sistemas de Arquivos - Linux

Segundo Machado e Maia (2007), o sistema de arquivos no Linux é baseado numa estrutura de diretórios hierárquica, sendo o diretório raiz (root) representado por uma (/). Veremos essa estrutura hierárquica com maior profundidade na Seção 3.2. Os sistemas de arquivos suportados pelas distribuições do Linux são (SILVA, 2018):

- EXT2: Second Extended File System ou EXT2, estava presente nas primeiras versões do Linux, mas era pouco eficiente.
- EXT3: o sistema EXT3 é uma versão do EXT2, e se tornou popular, estável e robusto por possuir um suporte eficiente, conhecido como journaling. Journaling é um recurso que gerencia a manipulação de arquivos. Journals (ou logs) registram nos discos as alterações realizadas nos arquivos antes de serem gravadas, e, por meio desse log, caso aconteça algum erro, como queda de energia ou o travamento inesperado, as operações podem ser recuperadas.
- EXT4: é a versão mais recente do EXT, apresentando um bom desempenho com a manipulação de arquivos grandes, porém ainda instável.
- ReiserFS: foi criado recentemente, possui suporte journaling e uma boa performance, além de ser suportado pelas versões do Linux.

Os sistemas EXT3 e ReiserFS são os mais utilizados pelas distribuições do Linux. Além destes sistemas existem outros suportados pelo Linux, como SWAP, XFS, dentre outros.

%=

Exemplificando

O Linux oferece suporte a vários sistemas de arquivos antigos, como MINIX, MS-DOS e EXT2. Também suporta os novos sistemas de arquivos com registro de mudança, como EXT3, JFS e ReiserFS, além do suporte a sistemas de arquivos criptográficos, como CFS, e a sistemas de arquivos virtuais, como /proc.

JONES, M. Anatomia do Sistema de Arquivos do Linux. 30 out. 2007. Disponível em: https://www.ibm.com/developerworks/br/library/l-linux-filesystem/index.html. Acesso em: 9 mai. 2018.

Sistemas de Arquivos - Windows

Os sistemas de arquivos do Windows são o FAT16, FAT32 e NTFS. O nome FAT significa *File Alocation Table*.

Segundo Tanenbaum (2003), o FAT16, usado no MS-DOS, utiliza endereços de memória de 16 bits e não suporta discos maiores que 2GB, gerando desperdício de espaço. Já o FAT32 utiliza endereços de memória de 32 bits, mas não reconhece arquivos maiores que 4GB.

O NTFS (New Technology File System) é o atual sistema de arquivos do Windows. O NTFS é seguro, possui suporte a discos maiores, além de implementar permissões de acessos aos arquivos. A partir do Windows 8, foi criado o sistema de arquivos ReFS (Resilient File System), com o objetivo de verificar e corrigir automaticamente os dados e manipular um grande volume de dados.



Qual sistema operacional é melhor ao trabalhar com arquivos? O Windows, com o padrão NTFS, ou o Linux, com o padrão Ext4?

Sem medo de errar

Agora que você já conheceu como os arquivos são usados pelo usuário, suas características, como se dá a nomeação e a estrutura de arquivos, os tipos de arquivos suportados pelos sistemas operacionais, como é realizado o acesso aos arquivos, os atributos dos arquivos e as operações realizadas em arquivos, vamos voltar ao nosso contexto. Na empresa XYZ Ltda., os arquivos de cadastros de clientes de todas as filiais estão armazenados em caixas, num depósito na matriz, localizada em São Paulo. Uma parte deste cadastro foi armazenada em CDs-ROM e outra parte em HD. Quais levantamentos devem ser feitos antes de iniciar os trabalhos com os arquivos da empresa XYZ Ltda.? Qual sistema de arquivos será implementado nos sistemas operacionais Linux e Windows? Como serão definidos os nomes, os atributos, o acesso e as operações dos arquivos da empresa XYZ Ltda.? Antes de definir o sistema de arquivos a ser usado, é necessário fazer um levantamento dos documentos que estão armazenados no depósito, verificando a estrutura destes arquivos. Como os arquivos ficam quardados em caixas, é necessário antes migrar estes dados para um banco de dados que irão gerenciar melhor estas informações. Após isso, é necessário definir o sistema de arquivo a ser utilizado. Na empresa XYZ Ltda. são utilizados os sistemas operacionais Linux e Windows. Como sistema de arquivos do Windows, podemos usar o padrão adotado atualmente, o NTFS, que é seguro e manipula um grande volume de dados. Para o Linux, muitas distribuições, ao serem instaladas, já trazem o sistema de arquivos instalado. Na maioria das distribuições, o sistema que vem instalado é o EXT4, que não é a versão atual, é seguro, porém instável, e atenderá bem às necessidades da empresa XYZ Ltda. Como a empresa trabalha por filiais, podemos sugerir que os nomes dos arquivos comecem identificando-as, conforme apresentado no Quadro 3.3. Por exemplo, a empresa possui o documento "Levantamento de Vendas Mensais", que contém informações de todas as vendas realizadas no mês por setor. Esse arquivo pode ser nomeado iniciando pela filial, o nome do setor, o tipo de vendas (diário, semanal, mensal, anual) e o mês/ano do arquivo. É necessário salientar que o Linux faz diferenciação de letras maiúsculas e minúsculas, e, durante a definição dos nomes, essa questão deve ser levada em consideração.

Quadro 3.3 | Sugestão de nomes de arquivos

Documento	Sugestão de nome		
Levantamento de Vendas Mensais	BH_Vendas_Mensais_05/2018 RJ_ Vendas_Mensais_01/2018		
Clientes de Pré-vendas anuais	SP_Pre_Vendas_2018		

Fonte: elaborado pela autora.

Quanto aos atributos, será necessário definir:

- Quem terá acesso a sistema por exemplo, o setor de Prévendas tem acesso somente aos arquivos do seu setor. Já o setor de vendas terá acesso aos seus arquivos e aos arquivos do setor de Prévendas.
- Inserção de senha para acesso a todos os arquivos.
- Identificação de quem criou o arquivo.
- Tamanho do arquivo.

- Data da criação do arquivo.
- Data da modificação do arquivo.

O acesso aos arquivos pode ser do tipo indexado, para acessar de forma rápida o registro desejado no arquivo. É necessário definir alguns parâmetros para que as operações sejam realizadas.

Quadro 3.4 | Sugestão de operações a serem realizadas.

Operação	Permissão
Criar	Todos os usuários podem criar seus próprios arquivos.
Deletar	Somente quem criou terá a permissão para deletar o arquivo.
Abrir	Somente usuários autorizados pelo setor poderão abrir o arquivo.
Ler	Todos os usuários poderão abrir o arquivo.
Gravar	Somente quem criou terá a permissão para gravar dados no arquivo.

Fonte: elaborado pela autora.

Avançando na prática

Métodos de acesso aos arquivos

Descrição da situação-problema

Durante a organização dos arquivos da empresa XYZ Ltda., o gerente de vendas questiona de qual maneira os dados dos arquivos serão acessados, uma vez que existem várias filiais e o volume de dados movimentados é grande. Existem os métodos de acesso sequencial, direto e indexado (ou por chave). O que levar em conta na escolha do método de acesso ao arquivo? Qual é o melhor método de acesso para resolver essa questão?

Resolução da situação-problema

A empresa XYZ Ltda. possui um volume elevado de dados sendo necessário escolher um método robusto de acesso aos arquivos. O método sequencial não permite o acesso direto ao arquivo, o que dificulta o acesso a informação. O método direto permite a leitura da posição do registro, porém, existe uma limitação: o arquivo deve ser definido com registros de tamanho fixo. O acesso indexado, ou por chave, é o indicado

para a empresa XYZ Ltda., por ser o método mais sofisticado. Quando uma aplicação deseja ter acesso ao registro, é necessário informar a chave do registro. É feita uma pesquisa no sistema de arquivos por meio da chave para se ter acesso direto ao registro do arquivo.

Faça valer a pena

1.



Na prática, um sistema de arquivo (*file system*, do inglês) é um conjunto de estruturas lógicas, ou seja, feitas diretamente via software, que permite ao sistema operacional ter acesso e controlar os dados gravados no disco. No universo Windows, o número de sistemas de arquivos é mais limitado. Na época do Windows 95, a Microsoft usava o sistema de arquivos FAT16. Devido às suas limitações, foi substituído pelo FAT32 que, anos depois, foi substituído pelo NTFS. Este é usado até hoje e se estabeleceu devido à flexibilidade. Já no amplo universo Linux, onde é possível encontrar uma enorme variedade de distribuições, o leque de sistemas de arquivos é bem maior. Os mais usados são o EXT3 e o EXT4, bem como o ReiserFS.

ALENCAR, F. Entenda o que é sistema de arquivos e sua utilidade no PC e no celular. **TechTudo.** 26 fev. 2016. Disponível em: http://www.techtudo.com.br/dicas-e-tutoriais/noticia/2016/02/entenda-o-que-e-sistema-de-arquivos-e-sua-utilidade-no-pc-e-no-celular.html>. Acesso em: 7 mai. 2018.

Sobre os sistemas de arquivo Linux e Windows, marque a alternativa correta.

- a) O Windows consegue reconhecer outros sistemas de arquivos além do NTSF.
- b) No Linux, o sistema de arquivos é estruturado em formato de árvore.
- c) O sistema de arquivo do Windows é case *sensitive*, ou seja, diferencia letras minúsculas e maiúsculas.
- d) O Linux consegue suportar uma quantidade limitada de sistemas de arquivos.
- e) O sistema FAT32 suporta diversos tipos de usuários do Windows.

2.

Nos sistemas *Linux*, tudo o que pode ser manipulado pelo sistema operacional é representado sob a forma de arquivo, incluindo diretórios, dispositivos e processos. Se tudo é arquivo, como o sistema operacional faz a distinção entre arquivo regular, diretório, dispositivo, processo?"



PAULO, D. R. **Fundamentos do sistema Linux - arquivos e diretórios.** 28 out. 2005. Disponível em: https://www.vivaolinux.com.br/artigo/Fundamentos-do-sistema-Linux-arquivos-e-diretorios/>. Acesso em: 09 mai. 2018.

A seguir, analise as asserções:

Todo arquivo tem cabeçalho, que contém informações, como tipo, tamanho, data de acesso e de modificação.

Porque

Por meio do cabeçalho, o GNU/Linux sabe quando se trata de arquivo regular, diretório, dispositivo, e processo.

Com relação às duas asserções assinale a alternativa correta:

- a) As asserções I e II são proposições verdadeiras, e a II é uma justificativa da I.
- b) As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa da I.
- c) A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.
- d) A asserção I é uma proposição falsa, e a II é uma proposição verdadeira.
- e) As asserções I e II são proposições falsas.

3.

Segundo a Microsoft, o ReFS foi criado para atender as novas necessidades de armazenamento do Windows, sendo elas a capacidade de lidar com grandes volumes, a resistência ao corrompimento de arquivos e o compartilhamento de conjuntos de armazenamento entre máquinas diferentes acessando diretamente o arquivo.



HAMMERSCHMIDT, R. Microsoft revela mais detalhes sobre o ReFS, seu novo sistema de arquivos. **Tecmundo**. 17 jan. 2012. Disponível em: https://www.tecmundo.com.br/windows-8/17975-microsoft-revelamais-detalhes-sobre-o-refs-seu-novo-sistema-de-arquivos.htm>. Acesso em: 10 mai. 2018.

Considerando as características do sistema de arquivos ReFs da Microsoft, escolha a opção correta:

- a) O sistema de arquivos ReFS foi desenvolvido do zero, não se baseando nos sistemas de arquivos existentes.
- b) A resiliência do ReFS está diretamente ligada a verificar se o arquivo está corrompido e recuperá-lo.
- c) O sistema de arquivos ReFS não é compatível com o NTFS, e caso a versão do Windows seja diferente, é necessário instalar o Windows 8 para que ele funcione.
- d) O sistema de arquivos ReFS é mais moderno, porém não suporta volumes grandes de dados nos arquivos.
- e) O ReFS, por suportar a otimização em tempo real de arquivos, prejudica a performance do sistema tornando-o lento.

Seção 3.2

Sistemas de diretórios

Diálogo aberto

Caro aluno, o sistema de arquivos contém diretórios e pastas que gerenciam os arquivos. Normalmente, os atuais sistemas de arquivos permitem que os usuários criem vários diretórios e subdiretórios, apresentando uma estrutura organizada dos arquivos. Por exemplo, o diretório "/" do Linux representa a raiz do sistema de arquivos e os demais diretórios estão abaixo dele, como arquivos de sistemas, arquivos de usuário, arquivos de aplicações, entre outros. No Windows, o diretório C:/ representa a raiz do sistema. Nesta seção você conhecerá como os diretórios são usados pelo usuário e suas propriedades, como funcionam os sistemas de diretório simples e hierárquico, como se dá a nomeação dos caminhos de diretórios e as operações realizadas em diretórios. Relembrando nosso contexto, a empresa XYZ Ltda. do ramo de seguros automotivos e residenciais, com filiais em todo o país, deseja organizar os dados e implementar o seu sistema de arquivos e, para isso, lhe contratou, já que você é um engenheiro da computação. Na empresa XYZ Ltda., os arquivos dos setores de pré-vendas e vendas são acessados durante todo o dia pelos funcionários dessas áreas para traçarem planos de ação das equipes de vendas. Porém, não existem diretórios específicos definidos para cada setor armazenar seus arquivos e diretórios do próprio sistema operacional. Desta forma, todos os usuários da empresa podem acessá-los, o que compromete a segurança dos dados, uma vez que, caso algum usuário acesse um diretório indevidamente, ele poderá altera-lo e/ou exclui-lo. Na empresa XYZ Ltda., já está definido o sistema de arquivos a ser usado tanto no Linux (EXT4) como no Windows (NTFS), além dos nomes, atributos, o acesso e as operações a serem realizadas nos arquivos. Como você fará a organização de diretórios? Como serão definidas as permissões de diretórios para os setores de vendas e pré-vendas? Todas essas questões darão base para você elaborar o anteprojeto de um sistema de proteção de arquivos. Para que você consiga responder a esses e outros questionamentos sobre os diretórios, nesta seção vamos conhecer mais sobre eles e os conteúdos pertinentes a este tema. Vamos lá?

Bons estudos!

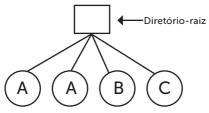
Não pode faltar

Segundo Machado e Maia (2007), a estrutura de diretórios é a organização lógica dos arquivos, em disco. O diretório é uma estrutura de dados que possuem entradas associadas aos arquivos, sendo que cada entrada possui os atributos de localização do arquivo, nome, dono, organização, dentre outros. Em alguns sistemas operacionais, como o Linux, todos os diretórios e pastas são considerados arquivos.

Diretórios Simples

Uma forma simples de sistema de diretório é manter um diretório contendo todos os arquivos, chamado de diretório raiz (TANENBAUM, 2003). Esse formato era utilizado nos primeiros computadores pessoais por haver apenas um usuário e, como vantagem, apresenta a simplicidade e a capacidade de encontrar rapidamente os arquivos. A Figura 3.3 apresenta um exemplo de sistema de diretórios com quatro arquivos de três usuários diferentes. A. B. e. C.

Figura 3.3 | Sistema de Diretórios Simples.



Fonte: Tanenbaum (2003, p. 296).

Sistemas de Diretório Hierárquico

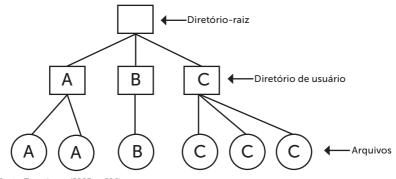
Segundo Tanenbaum (2003), o problema existente em um sistema com vários usuários é que pode ocorrer de diferentes

usuários nomearem seus arquivos com os mesmos nomes. Por isso, esse esquema não é mais utilizado em sistemas multiusuário.

Para evitar conflitos causados por diferentes usuários escolhendo o mesmo nome, é necessário dar um diretório privado para cada um. Assim, os nomes escolhidos por um usuário não interfeririam nos nomes escolhidos por outro, podendo existir arquivos com o mesmo nome em dois ou mais diretórios. Esse esquema é chamado de sistema de diretórios em dois níveis.

A Figura 3.4 apresenta um sistema de diretórios em dois níveis, em que cada usuário possui seu diretório. Assim, o sistema, conhecendo o usuário, sabe em qual diretório buscar o arquivo.

Figura 3.4 | Sistema de Diretórios em dois níveis



Fonte: Tanenbaum (2003, p. 296).

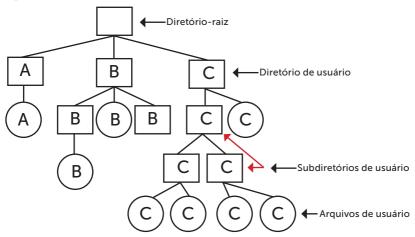
O esquema de diretórios em dois níveis elimina o conflito de nomes entre usuários, porém, não é adequado para usuários com um grande número de arquivos. Por exemplo, um programador de sistemas pode ter:

- Um conjunto de arquivos de códigos, que juntos formam um sistema;
- Outro conjunto de arquivos referentes a projetos já entregues;
- Outro conjunto de arquivos tratando de livros específicos da área, e assim por diante.

Dessa forma, faz-se necessário agrupar esses arquivos segundo as necessidades do usuário. Assim, é necessária uma hierarquia geral ou árvore de diretórios (folder tree) para permitir que os usuários tenham tantos diretórios quanto precisarem para agrupar

os seus arquivos. A Figura 3.5 apresenta a estrutura de diretórios hierárquicos. Nela, os diretórios A, B e C pertencem a usuários diferentes, e dois desses usuários criaram subdiretórios para organizar os seus arquivos. Quase todos os sistemas de arquivos modernos são organizados com essa estrutura, por proporcionar uma melhor organização dos trabalhos.

Figura 3.5 | Sistema de Diretórios Hierárquicos



Fonte: Tanenbaum (2003, p. 297).



Para saber mais sobre o gerenciamento de arquivos e diretórios do Linux acesse:

SHIELDS, I. Gerenciamento de arquivo e diretório. 10 nov. 2009. Disponível em: https://www.ibm.com/developerworks/br/linux/library/l-lpic1-v3-103-3/index.html. Acesso em: 14 mai. 2018.

Nomes de Caminhos dos Diretórios

Segundo Tanenbaum (2003), quando o sistema de arquivos é organizado por meio de uma árvore de diretórios, é necessário definir uma forma de especificar o nome dos arquivos. Para isso, são usados dois métodos: nome de caminho absoluto e nome de caminho relativo.

O método <u>nome de caminho absoluto</u> é formado pelo caminho entre o diretório-raiz e o arquivo. Os nomes de caminhos absolutos sempre iniciam no diretório-raiz e são únicos. Como exemplo, o caminho /usuário/meus_documentos/atividades.txt significa que dentro do diretório-raiz há um diretório chamado "usuário", dentro do diretório "usuário" existe um subdiretório chamado "meus_documentos" e dentro do subdiretório "meus_documentos" existe um arquivo chamado "atividades.txt". No Windows, os componentes do caminho são separados por \. No Unix, são separados por /.

O método <u>nome de caminho relativo</u> é usado em conjunto com o conceito de diretório atual ou diretório de trabalho. Desta forma, é possível que o usuário designe um diretório como o diretório atual de trabalho, em que todos os nomes de caminhos não comecem no diretório-raiz.

Como exemplo, se o diretório atual for /usuário/meus_documentos/, o arquivo cujo caminho absoluto for /usuário/meus_documentos/atividades pode ser referenciado apenas como atividades



Analise o endereço do seguinte projeto de um sistema hospitalar alocado num servidor Linux: /home/user/projetos/sistema_hospitalar. O caminho absoluto do endereço é /home/user/projetos/sistema_hospitalar e o caminho relativo do endereço é sistema_hospitalar, se estivermos no diretório /home/user/projetos.

Ainda, segundo Tanenbaum (2003), cada processo possui seu próprio diretório de trabalho, e caso ocorra alguma alteração, nenhum outro processo é afetado e nenhum vestígio da mudança é deixado no sistema de arquivos. Procedimentos de biblioteca (conjunto de subprogramas que auxiliam no desenvolvimento do software) raramente alteram o diretório de trabalho, pois se alterarem o diretório de trabalho e não retornarem aonde estavam quando finalizarem, o resto do programa poderá não funcionar.

Os sistemas operacionais que suportam um sistema de diretório hierárquico possuem duas entradas especiais em cada diretório: "." (ponto), que se refere ao diretório atual, e ".." (ponto-ponto), referindo-se a seu pai.

Por exemplo, se um usuário estiver trabalhando no diretório / usuário/meus_documentos/ e utilizar o ponto-ponto, ele estará subindo na árvore de diretórios, ou seja, referenciando o diretório /usuário. Caso ele utilize o ponto, estará referenciando o próprio diretório (/usuário/meus_documentos/).

Operações com Diretórios

As chamadas aos sistemas que gerenciam as operações com diretórios variam de sistema para sistema. Segundo Tanenbaum (2003), as principais operações de manipulação de diretórios são:

- <u>Create</u>: cria um diretório. Um diretório é criado sem conteúdo, ou seja, vazio.
- <u>Delete</u>: apaga um diretório. Somente um diretório vazio pode ser apagado.
- Opendir: lê um diretório. Antes de ser lido, um diretório precisa ser aberto.
- <u>Closedir</u>: após a leitura do diretório, o mesmo precisa ser fechado, liberando, assim, espaço em disco.
- Rename: permite a troca do nome de um diretório.



Seguem as principais operações realizadas nos diretórios dos sistemas operacionais Linux e Windows.

Linux:

- Criar: mkdir EXEMPLO (cria o diretório EXEMPLO no diretório atual do usuário).
- Deletar: rmdir EXEMPLO (apaga o diretório EXEMPLO se ele estiver vazio).
- Abrir: cd EXEMPLO (abre o diretório EXEMPLO a partir do diretório atual do usuário).

- Fechar: closedir EXEMPLO (fecha o diretório EXEMPLO).
- Renomear: mv EXEMPLO TESTE (renomeia o diretório EXEMPLO para TESTE).

Windows

- Criar: clique com o botão direito do mouse no local que deseja criar um novo diretório e, em seguida, na opção "Novo -> Pasta".
- Deletar: selecione o diretório que deseja excluir e clique com o botão direito do mouse na opção "Excluir".
- Abrir: basta dar um duplo clique no diretório.
- Fechar: clique no X do diretório para fechá-lo.
- Renomear: selecione o diretório que deseja renomear e clique com o botão direito do mouse na opção "Renomear".

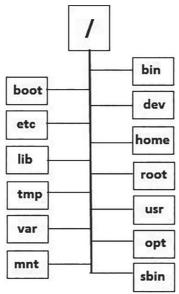


Vimos nesta seção as principais operações realizadas com diretórios nos sistemas operacionais. Além dessas operações, podemos citar como exemplo a operação Link, que permite a um arquivo aparecer em mais de um diretório. Essa chamada ao sistema especifica um arquivo existente e um nome de arquivo, criando uma ligação do arquivo existente com o nome especificado pelo caminho (TANENBAUM, 2003).

Sistemas de Diretórios do Linux

No Linux, o sistema de diretórios é baseado numa estrutura hierárquica e os diretórios são implementados por meio de arquivos. As partições e discos fazem parte do diretório-raiz ou "/". Dentro do diretório-raiz existem diretórios que possuem funções distintas. A Figura 3.6 apresenta a estrutura de diretórios do Linux

Figura 3.6 | Sistema de Diretórios do Linux.



Fonte: elaborada pela autora

A seguir, apresentamos as definições de cada diretório que estão abaixo do diretório-raiz (JUNIOR, 2018):

- /bin: contém aplicativos e utilitários do sistema.
- /boot: contém todos os arquivos necessários para a inicialização do sistema e o gerenciador de boot.
- /etc: contém todos os arquivos de configuração do sistema.
- /dev: contém todos os dispositivos do sistema, tais como: modem, HD, floppy disk, CD-ROM, entre outros.
- /home: contém diretórios e arquivos de trabalho dos usuários comuns do sistema.
- /lib: contém as bibliotecas compartilhadas e módulos do kernel.
- /root: contém diretórios e arquivos de trabalho do superusuário (root).
- /usr: contém os programas de usuários, sistemas de janelas, jogos, entre outros.
- /tmp: contém os arquivos temporários utilizados por algumas aplicações.

- /var: contém os arquivos de dados variáveis, como log do sistema e diretórios de spool.
- /mnt: diretório reservado para montagem de dispositivos e sistemas de arquivos em partições e discos locais e de rede.
- /opt: diretório reservado para instalação de aplicações de terceiros, como OpenOffice, softwares pagos, dentre outros.
- /sbin: contém ferramentas de configuração e manutenção do sistema.

Segundo Machado e Maia (2007), a localização de um arquivo dentro da estrutura de diretórios é indicada por um pathname, que pode ser dos tipos absoluto ou relativo. Um pathname absoluto indica a localização do arquivo por meio do caminho completo do diretório. Já um relativo é representado pelo nome parcial do diretório. Ainda segundo os mesmos autores, um sistema de arquivos proporciona um mecanismo de compartilhamento chamado link (ligação). Um link é a entrada de um diretório que referencia um arquivo em outro diretório. Um arquivo pode ter vários links. Com a utilização de links existe uma redução de espaço em disco, a possibilidade de compartilhamento entre diversos usuários, dentre outros.

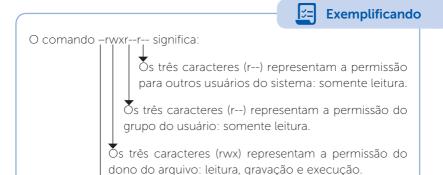
Todo arquivo ou diretório do Linux tem um dono (*user*) e pertence a um grupo (*group*). Para cada categoria de usuário existem três tipos de acesso: leitura (r), gravação (w) e execução (x). O Quadro 3.5 apresenta as permissões (privilégios) que os arquivos e diretórios possuem, de acordo com o tipo de acesso.

Quadro 3.5 | Permissões de arquivos e diretórios

Permissão	Arquivos Diretórios		
r	Leitura e cópia	Lista conteúdo	
W	Gravação	Cria, elimina e renomeia	
Х	Execução	Acessa os arquivos	

Fonte: elaborado pela autora.

Existe ainda a definição de quem pode acessar os diretórios: somente o dono, todos os usuários, somente usuários do mesmo grupo ou qualquer conjunto de usuários.



O primeiro caractere informa se é um arquivo ou diretório.

Sistemas de Diretórios do Windows

Segundo Tanenbaum (2003), o sistema de arguivos NTFS é baseando numa estrutura hierárquica. No Windows, o sistema de arquivos está concentrado em pastas ou diretórios C:, D:, dentre outras.

A Figura 3.7 apresenta a estrutura de diretórios do Windows. O diretório C: contém as pastas "Usuários", "Arquivos de Programas" e "Windows". A pasta "Usuários" contém as principais pastas utilizadas pelos usuários. A pasta "Arquivos de Programas" é onde estão concentrados todos os programas instalados no computador. A pasta "Windows" é a mais importante, por nela constar toda a estrutura do sistema operacional.

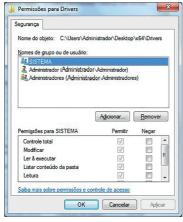
C: Arquivos de Programas Usuários Internet Explorer Área de Trabalho Microsoft Office **Downloads** Mozilla Firefox Documentos **PDFCreator** Imagens Kaspersky Windows Vídeos

Figura 3.7 | Sistema de Diretórios do Windows.

Fonte: elaborada pela autora.

No Windows, o separador de nomes é representado por "\" (contra barra) e também existe o conceito de caminho absoluto e relativo, como no Linux. Além disso, é possível inserir permissões de acesso para cada usuário (administrador do sistema e usuário comum). Para isso, basta configurar as permissões na pasta desejada. A Figura 3.8 apresenta a pasta Drivers e as permissões que o usuário SISTEMA possui. Observe que o usuário SISTEMA possui as permissões de controle total, modificar, ler e executar, listar conteúdo de pastas, leitura, gravar e permissões especiais.

Figura 3.8 | Sistema de Diretórios do Windows.



Fonte: captura de tela do Windows, elaborada pela autora.



Na sua opinião, a hierarquia de diretórios do Linux é mais complexa do que a do Windows? Por quê?

Sem medo de errar

Agora que você já conheceu como os diretórios são usados pelo usuário, suas propriedades e as operações que podem ser realizadas, vamos voltar ao nosso contexto. Na empresa XYZ Ltda., os arquivos dos setores de pré-vendas e vendas são acessados durante todo o dia, porém, não existem diretórios específicos definidos para

cada setor armazenar seus arquivos e diretórios do próprio sistema operacional. Isso pode comprometer a segurança, pois caso um usuário acesse um diretório indevidamente, poderá altera-lo e/ou excluí-lo. O sistema de arquivos já foi definido tanto no Linux (EXT4) como no Windows (NTFS), além dos nomes, os atributos, o acesso e as operações a serem realizadas nos arquivos. Como você fará a organização de diretórios? Como seriam definidas as permissões de diretórios para os setores de vendas e pré-vendas?

Um diretório fornece acesso aos arquivos e a outros diretórios vinculados a ele. A questão da falta de estrutura dos diretórios da empresa XYZ Ltda. pode ser em função da má implementação de diretórios realizada anteriormente. Assim, é necessário analisar a estrutura atual e reorganizar os diretórios e arquivos, implementando e relacionando cada entrada do diretório aos arquivos que estão armazenados nele. Um fator importantíssimo é que os diretórios do próprio sistema operacional estão acessíveis a todos os usuários. Isso não pode acontecer. Logo, os diretórios devem ser reestruturados para que somente os administradores do sistema tenham acesso a eles.

A empresa XYZ Ltda. possui os sistemas operacionais Linux e Windows, e a organização de diretórios para ambos pode ser representada pelo exemplo da Figura 3.9. A organização de diretórios pode ser definida pelo administrador do sistema ou pelo próprio usuário.

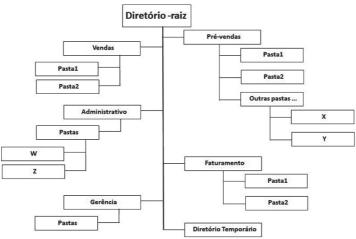


Figura 3.9 | Sistema de Diretórios do Windows.

Fonte: elaborada pela autora.

Na Figura 3.9 é apresentada uma organização estruturada de diretórios. Dentro do diretório-raiz são encontrados os diretórios Vendas, Pré-vendas, Administrativo, Faturamento, Gerência e Diretório Temporário. Em cada diretório há subdiretórios criados pelos usuários.

Será necessário definir as permissões de acesso dos setores de vendas e pré-vendas para que cada setor consiga operar seus diretórios de forma eficiente. O setor de pré-venda terá acesso para gravar, ler e executar arquivos, ou seja, terá permissão total ao seu diretório de arquivos, porém, será permitido apenas ler os arquivos do diretório de vendas. Já o setor de vendas, terá acesso total ao seu diretório, podendo ler e executar dados do setor de prévendas. Uma vez definidas as permissões de acesso de cada setor, as operações permitidas para cada setor são:

Quadro 3.6 | Definição das permissões de diretórios para o setor de vendas e pré-vendas

Permissão	Ler	Gravar	Executar
Pré-vendas	Χ	Χ	Χ
Vendas	Χ	Χ	Χ
Acesso aos arquivos de outro setor	Χ	-	-

Fonte: elaborado pela autora.

Avançando na prática

Limite de arquivos em um diretório

Descrição da situação-problema

A empresa XYZ Ltda. possui um grande volume de dados, que são armazenados em diretórios e pastas. O diretor da área de prévendas identificou que o diretório de armazenamento das propostas de vendas mensais não suporta um armazenamento superior a 50000 arquivos. Além disso, o diretor relatou que ao gravar o arquivo "Conexão de Vendas" com o nome "con", o sistema não permitiu a sua criação. Diante do exposto, os seguintes questionamentos são feitos a você: a quantidade de arquivos é limitada num diretório? Caso a quantidade de arquivos em um diretório seja limitada, quem gerencia é o sistema operacional ou o sistema de arquivos? Por que a criação de diretórios com determinados nomes não é possível?

Resolução da situação-problema

Existe um limite para a quantidade de arquivos em um diretório. Este limite é controlado pelo sistema de arquivos. Para verificar a quantidade de arquivos suportada num diretório, verifique qual é o sistema de arquivo utilizado. O sistema de arquivos NTFS e o EXT4 suportam mais de 4 bilhões de arquivos numa pasta ou diretório. Alguns nomes são utilizados nos sistemas operacionais, como palavras reservadas, ou seja, exclusivas para uso do sistema. Logo, não é possível criar arquivos com esses nomes. Por exemplo, a criação de um arquivo com o nome "con" é barrada pelo sistema, pois "con" vem de console, e este é um comando que prepara a entrada de dados via teclado.

Faça valer a pena

1.



Árvore de diretórios ou *folder tree* é a forma como as pastas dos sistemas Linux estão organizadas. Elas têm uma hierarquia, para facilitar a organização do sistema, seus arquivos, bibliotecas e inclusive para melhorar a segurança do sistema.

SANTOS, T. A Arvore de diretórios do Linux. 13 out. 2016. Disponível em: https://fsocietybrasil.org/a-arvore-de-diretorios-do-linux/. Acesso em: 11 mai. 2018.

Sobre o sistema de diretório hierárquico do Linux, marque a alternativa correta.

- a) No primeiro nível, estão os "diretórios de Usuário", usados pelo HOME e Kernel do sistema.
- b) A partir do diretório /usr começa a terceira hierarquia dos sistemas de arquivos Linux.
- c) Na segunda hierarquia, ficam funcionalidades privadas /usr/local.
- d) /tmp diretório onde estão armazenados os arquivos temporários, que não são deletados ao iniciar o sistema.
- e) /boot diretório onde estão os arquivos usados na inicialização do sistema.

2.



Quando usuários acostumados com o Windows se deparam pela primeira vez com o sistema Linux, uma das confusões que aparecem é com relação a diferença entre os sistemas de arquivos implementado pela Microsoft e pelo Linux. COTA, A. Diferenças entre o sistema de arquivos do Windows e Linux. Disponível em: https://www.vivaolinux.com.br/artigo/Diferencas-entre-o-sistema-de-arquivos-do-Windows-e-Linux. Acesso em: 11 mai. 2018.

A seguir, analise as asserções:

No Linux, os discos e partições não aparecem necessariamente como unidades diferentes, como o C:, D: e E: do Windows.

POR QUE

Tudo faz parte de um único diretório, chamado diretório-raiz. Assim, de acordo com tipo de arquivo, são direcionados para os diretórios específicos.

Com relação às duas asserções assinale a alternativa correta:

- a) As asserções I e II são proposições verdadeiras, e a II é uma justificativa da I.
- b) As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa da I.
- c) A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.
- d) A asserção I é uma proposição falsa, e a II é uma proposição verdadeira.
- e) As asserções I e II são proposições falsas.
- **3.** O caminho absoluto é o nome completo do caminho e o caminho relativo é o nome parcial do caminho. Imagine que você tenha criado um projeto de um sistema web no caminho C:\usuarios\carlos.pereira\Projetos\SistemaWeb\Paginas\Index.html.

Qual é o nome do caminho absoluto e qual o nome do caminho relativo para o projeto do sistema web?

Assinale a alternativa correta.

- a) Caminho absoluto: C:\usuarios\carlos.pereira\Projetos\SistemaWeb\ Paginas\Index.html; e caminho relativo: \Paginas\Index.html.
- b) Caminho absoluto: $\SistemaWeb\Paginas\Index.html;$ e caminho relativo: C:\usuarios\carlos.pereira\Projetos\SistemaWeb\Paginas.
- c) Caminho absoluto: C:\usuarios\carlos.pereira\; e caminho relativo: \Projetos\SistemaWeb\Paginas\Index.html.
- d) Caminho absoluto: C:\usuarios\carlos.pereira\Projetos; e caminho relativo: \Paginas\Index.html.
- e) Caminho absoluto: C:\usuarios\carlos.pereira\Projetos\SistemaWeb\Paginas\Index.html; e caminho relativo: \SistemaWeb\Paginas\Index.html.

Seção 3.3

Implementação e segurança de sistemas de arquivos

Diálogo aberto

Caro aluno, existem vários métodos de implementação do sistema de arquivos e diretórios, e nesta seção conheceremos alguns deles. Por exemplo, o Netflix possui um sistema de arquivos onde são armazenados os filmes e séries que são exibidos por streaming ao cliente. Além disso, possui um algoritmo de recomendação que determina as sugestões de filmes e séries para cada usuário. Nesta seção, você conhecerá como é realizada a implementação do Sistema de Arquivos e os métodos de implementação de Arquivos e Diretórios, como é feito o gerenciamento de espaço em disco e a importância da segurança e confiabilidade do Sistema de Arquivos. Relembrando nosso contexto, você foi contratado como engenheiro da computação pela empresa XYZ Ltda., que atua no ramo de seguros automotivos e residenciais, com filiais em todo o país e que deseja organizar os dados e implementar um sistema de arquivos.

A empresa XYZ Ltda. é segmentada nas seguintes áreas: prévendas, responsável por captar os clientes que possuem um desejo de obter um seguro; vendas, área que efetiva o seguro do cliente; faturamento, que é responsável pelo faturamento de todas as filiais; e administrativo, responsável pela emissão e envio dos boletos aos clientes e pela comissão de cada funcionário de vendas. Ela precisa que os dados contidos em seus arquivos sejam protegidos para que somente as pessoas ou setores autorizados tenham acesso a eles. Dessa forma, surgem os seguintes questionamentos: qual método você utilizá na implementação de arquivos na empresa XYZ Ltda.? Como você fará o gerenciamento de espaço em disco destes arquivos? Quais são os mecanismos de proteção que podem ser implementados na empresa XYZ Ltda., considerando os sistemas operacionais Linux e Windows?

Para que consiga responder a esses e outros questionamentos sobre a implantação e segurança do sistema de arguivos, vamos

conhecer mais sobre eles, e você conseguirá elaborar um anteprojeto que deverá ser entregue para a avaliação dos gestores da empresa, apresentando todos os dados obtidos no trabalho realizado.

Bons estudos!

Não pode faltar

Para cada sistema operacional existem diferentes métodos de implementação de arquivos e diretórios, sendo importante levar em consideração como será realizado o controle do armazenamento dos arquivos em disco.

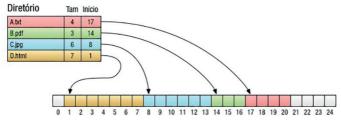
Implementação do Sistema de Arquivos

Segundo Machado e Maia (2007), o sistema operacional precisa controlar quais áreas ou blocos no disco estão livres quando um arquivo é criado. A seguir serão apresentados os principais métodos de implementação de arquivos utilizados nos sistemas operacionais.

Alocação Contígua: segundo Tanenbaum (2003), a implementação de arquivos por alocação contígua (Figura 3.10) é o método mais simples, em que os arquivos são armazenados de forma sequencial no disco. Assim, se você tem um disco rígido com blocos de tamanho 1MB, por exemplo, e um arquivo cujo tamanho seja de 40MB, você utilizará 40 blocos sequenciais de disco para alocar o arquivo, e assim por diante. A implementação desse método é simples e possui um bom desempenho, porém, como desvantagem, gera a fragmentação do disco, ou seja, os arquivos ficam espalhados por todo disco rígido do computador, o que compromete o desempenho, e ainda é necessário especificar o tamanho do arquivo durante a criação.

Na Figura 3.10 são apresentados quatro arquivos, com seus nomes, o tamanho dos blocos de disco e o respectivo bloco onde começará a alocação do arquivo. Por exemplo, o arquivo D.html inicia a alocação a partir do bloco de disco 1, com tamanho de 7 blocos de disco.

Figura 3.10 | Alocação Contígua

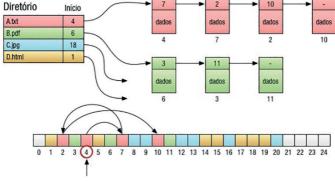


Fonte: Junior ([s.d.]).

Alocação por Lista Encadeada: segundo Machado e Maia (2007), outro método de armazenamento de arquivos consiste em organizar os mesmos como um conjunto de blocos ligados logicamente no disco, independentemente de sua localização física. Cada bloco contém um ponteiro para o bloco seguinte do arquivo. A Figura 3.11 apresenta a implementação de alocação por lista encadeada. Nela são apresentados quatro arquivos, com seus respectivos nomes e o bloco onde começará a alocação do arquivo.

Segundo Junior ([s.d.]), cada bloco do arquivo é composto por dados e pelo endereço do próximo bloco onde está a continuação dos dados. Por exemplo, o arquivo A.txt se inicia na posição 4 para o primeiro bloco e contém o endereço para o próximo bloco do arquivo, que é o bloco 7. O próximo bloco de dados do arquivo será armazenado no bloco 2. O bloco 2 faz referência para o próximo endereço de armazenamento, o bloco 10, e assim por diante.

Figura 3.11 | Alocação por Lista Encadeada



Fonte: Junior ([s.d.]).

Segundo Tanenbaum (2003), neste método não há desperdício de espaço com a fragmentação, além de precisar somente do primeiro endereço do disco para acessar todo o arquivo, pois os demais são encontrados a partir dele. A desvantagem é que o acesso aleatório é lento, além de gastar espaço de memória para armazenar os ponteiros.

Alocação por Lista Encadeada usando uma tabela na memória: segundo Tanenbaum (2003), nesse método cada palavra de ponteiro de cada bloco de disco é inserida em uma tabela na memória principal, chamada de FAT (*File Allocation Table*). A Figura 3.12 apresenta a implementação de alocação por lista encadeada usando uma tabela na memória, onde são apresentados quatro arquivos, com seus respectivos nomes e o bloco onde começará a alocação do arquivo. Cada arquivo aponta para uma entrada na tabela de memória, que, por sua vez, contém o endereço do próximo bloco onde está a continuação dos dados. Por exemplo, o arquivo A.txt aponta para a entrada 4 da tabela de memória, e esse endereço contém o endereço para o próximo bloco do arquivo, que é o bloco 7. A entrada 7 da tabela de memória faz referência para o próximo endereço de armazenamento, o bloco 2, e assim por diante.

(tabela na Diretório Início memória) A.txt 4 0 B.pdf 6 1 C.jpg 18 10 D.html 11 1 4 3 7 2 8 9 10 11

Figura 3.12 | Alocação por Lista Encadeada usando uma tabela na memória

Fonte: Junior ([s.d.]).

A vantagem desse método é que o acesso aleatório se torna mais fácil pela tabela estar carregada na memória. Porém, como desvantagem, a tabela deve estar na memória o tempo todo para funcionar.

Exemplificando

A desvantagem do método de alocação por lista encadeada usando uma tabela na memória é que a tabela deve estar na memória o tempo todo para funcionar. Por exemplo, para um disco de 20GB e blocos de 1KB, a tabela precisará de 20 milhões de entradas, uma para cada um dos 20 milhões dos blocos de disco, ou seja, a tabela pode ser muito grande para ser mantida em memória.

I-nodes: segundo Tanenbaum (2003), nesse método cada arquivo é associado a uma estrutura chamada I-node (index-node), relacionando os atributos e os endereços em disco dos blocos de arquivos. Com o I-node é possível encontrar todos os blocos de arquivos. Por exemplo, o arquivo A.txt foi associado a uma tabela I-node (Figura 3.13), com os seus atributos e endereços de disco, que fazem referência aos blocos de arquivos.

Diretório i-node dados dados A.txt Atributos B.doc bloco 4 bloco 7 ends de disco cont. bloco 8 dados bloco 15 bloco 3

Figura 3.13 | I-nodes

Fonte: Junior ([s.d.]).

Nesse método, a tabela precisa ser carregada somente quando o arquivo correspondente estiver aberto. Uma desvantagem é que se existirem arquivos que precisam crescer além do esperado, seria necessário ter dois ou mais endereços de disco apontando para outros blocos de disco cheios de endereços.



Você sabia que num determinado dispositivo de armazenamento o I-node pode ocupar até 1% do espaço total (LINUXANDO, 2018)? A quantidade de arquivos e diretórios influenciam sobre a quantidade de I-nodes?

Esse método é utilizado no sistema operacional Unix. No Linux, o método I-node armazena num disco rígido, por exemplo, as informações de permissões de acesso, identificação dos donos dos arquivos, o tamanho do arquivo e os ponteiros. Segundo Couto (2018), o núcleo do sistema operacional, ao ler um arquivo ou diretório, verifica as permissões contidas no I-node do arquivo e, caso sejam negadas, o usuário não terá acesso às suas informações. Somente após a verificação das permissões do I-node é que são validadas as permissões dos arquivos ou diretórios.

Pesquise mais

Cada entrada de registo de um I-node tem um tamanho de 128 bytes, e as informações armazenadas podem incluir: o número do I-node, o controle de acesso, as propriedades ou atributos, o número dos blocos do dispositivo (localização no disco), entre outros. Para saber mais sobre esse assunto, consulte este artigo:

LINUXANDO. **Compreender o Inode em Linux**. Disponível em: https://www.linuxando.com/tutorial.php?t=Compreender%20o%20 Inode%20em%20Linux_7>. Acesso em: 22 maio 2018.

Implementação de Arquivos e Diretórios

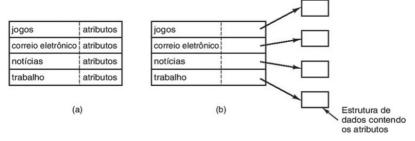
Segundo Tanenbaum (2003), para localizar a entrada de um diretório, o sistema operacional usa o nome do caminho do arquivo dado pelo usuário. A entrada de um diretório contém a informação

necessária para encontrar os blocos de disco, que pode ser: o endereço do disco do arquivo, o número do primeiro bloco, o número do I-node.

A função principal do sistema de diretório é mapear o nome do arquivo na informação necessária para localizar os dados.

Os atributos do arquivo (proprietário do arquivo e data da criação, por exemplo) são armazenados na entrada do diretório. A Figura 3.14 (a) apresenta um diretório com entradas de tamanho fixo, contendo um nome, os atributos e os endereços de discos. Os sistemas que utilizam I-nodes podem armazenar os atributos em I-nodes, assim, as entradas de diretórios conterão um nome de arquivo e um número I-node, conforme apresentado na Figura 3.14 (b).

Figura 3.14 | Atributos na entrada do diretório (a) e atributos em I-nodes (b)



Fonte: Tanenbaum (2003, p.305).

Ainda segundo Tanenbaum (2003), os sistemas operacionais modernos suportam nomes de arquivos mais longos e com tamanhos variáveis. Para fazer essa implementação, é necessário definir um limite de 255 caracteres para o tamanho do nome do arquivo. Essa implementação é simples, mas consome uma quantidade grande e desnecessária de espaço no diretório, uma vez que nem todos os nomes de arquivos são grandes.

Uma alternativa para esse método é definir a entrada do diretório como uma parte fixa a partir do tamanho da entrada dos arquivos (variável), seguida por dados de formato fixo (proprietário do arquivo, horário de criação, entre outros). A Figura 3.15 (a) apresenta um

diretório com dois arquivos ("a.out" e "projeto.c"), com o tamanho da entrada variável e os atributos do arquivo.

Outra forma de tratar nomes de arquivos variáveis é apresentada na Figura 3.15 (b). As entradas de diretório são de tamanho fixo e os nomes de arquivos ficam agrupados no final do diretório, em uma área chamada *heap*. Desta forma, os ponteiros dos arquivos apontam para início dos nomes dos arquivos no final do diretório.

tamanho da entrada do ponteiro para o arquivo 1 nome do entrada para atributos do o arquivo 1 arquivo 1 arquivo 1 atributos do arquivo 1 ponteiro para o 1/0 nome do tamanho da arquivo 2 entrada do atributos do arquivo 2 arquivo 2 atributos do entrada para arquivo 2 o arquivo 2 área temporária 10 0 p no final do 0 t e 0 diretório

C

(b)

0

Figura 3.15 | Atributos na entrada do diretório (a) e atributos em I-nodes (b)

Fonte: Junior ([s.d.]).

(a)

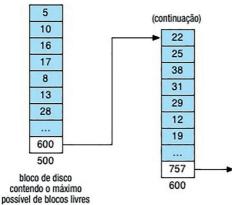
10

C

Gerenciamento de Espaço em Disco

O sistema operacional precisa gerenciar o espaço em disco dos blocos livres, garantindo um bom desempenho ao sistema. Segundo Tanenbaum (2003), são utilizados dois métodos para monitorar os blocos livres: lista encadeada de blocos e mapa de bits. No método da lista encadeada de blocos, cada bloco contem a quantidade de espaços livres que puderem ter e possuem a localização dos blocos livres, conforme apresentado na Figura 3.16. Por exemplo, um disco de 16GB precisa de uma lista livre de 16.794 blocos para conter 2^{24} número de blocos de disco.

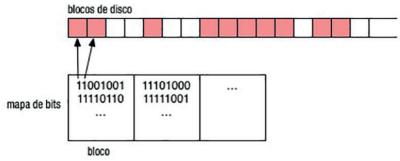
Figura 3.16 | Lista encadeada de blocos



Fonte: Junior ([s.d.]).

Segundo Deitel, Deitel e Choffnes (2005), um mapa de bits possui um bit para cada bloco físico do sistema de arquivos. O bit igual a 1 indica que o bloco está em uso, e o bit 0 que não está, conforme apresentado na Figura 3.17. Caso seja necessário alocar mais um bloco físico para o disco, é só percorrer o mapa de bits para encontrar o bit igual a 0.

Figura 3.17 | Mapa de Bits



Fonte: Junior ([s.d.]).

Ainda segundo Deitel, Deitel e Choffnes (2005), a vantagem do mapa de bits em relação à lista encadeada de blocos é que o sistema de arquivos consegue saber rapidamente onde há blocos livres, além de ser simples a sua implementação.

Segurança e Confiabilidade do Sistema de Arquivos

Segundo Machado e Maia (2007), os arquivos são compartilhados com usuários e utilizados para diversas finalidades. Assim, é necessário que o sistema operacional ofereça proteção desses arquivos para que usuários e processos que não tenham permissão consigam acessálos. Ainda segundo Machado e Maia (2007), um sistema de arquivos possui diferentes mecanismos de proteção, como:

- Senha de acesso: para ter acesso ao arquivo, é necessário que o usuário a conheça e que o sistema conceda o acesso ao arquivo. Nesse caso, como um arquivo possui apenas uma senha, não é possível definir quais tipos de operações serão realizadas.
- <u>Grupo de usuários:</u> essa proteção consiste em associar cada usuário a um grupo para compartilhar arquivos. Durante a criação do arquivo, o usuário define quais usuários terão acesso a ele.
- <u>Lista de controle de acesso (Access Control List ACL)</u>: é uma lista associada a cada arquivo, com as permissões de cada usuário. Quando um usuário tenta acessar um arquivo, o sistema operacional verifica sua permissão para autorizar ou não a operação realizada.

Segundo Tanenbaum (2003), existem questões de confiabilidade que devem manter a integridade dos arquivos, como cópias de segurança e consistência do sistema de arquivos. As cópias de segurança (backups) garantem que os dados sejam copiados e armazenados em diferentes lugares e dispositivos. A consistência do sistema de arquivos permite que os arquivos sejam salvos antes de finalizar sua tarefa. Assim, caso um usuário esteja modificando um arquivo e a luz acabe, por exemplo, a parte trabalhada até o momento é salva em disco.



Vimos nesta seção que existem diferentes mecanismos de proteção de um sistema de arquivos. A escolha do mecanismo dependerá do sistema operacional utilizado. Por exemplo, no Linux, os usuários e grupos de usuários são agrupados por atividades em comum e possuem acesso para realizarem essas tarefas.

Sem medo de errar

Agora que você já conheceu como é realizada a implementação do Sistema de Arquivos, os métodos de implementação de Arquivos e Diretórios, como é realizado o gerenciamento de espaço em disco e a importância da segurança e confiabilidade do Sistema de Arquivos, vamos voltar ao nosso contexto. A empresa XYZ Ltda. precisa que os dados contidos em seus arquivos sejam protegidos, para que somente as pessoas ou setores autorizados tenham acesso a eles. Assim, surgem os seguintes questionamentos: qual método será utilizado para a implementação de arquivos na empresa XYZ Ltda.? Como será feito o gerenciamento de espaço em disco desses arquivos? Quais são os mecanismos de proteção que podem ser implementados, considerando os sistemas operacionais Linux e Windows?

Existem vários métodos de implementação de arquivos. Neste caso, deve-se analisar qual método será melhor aplicado ao sistema operacional utilizado pelo usuário. A empresa XYZ Ltda. utiliza os sistemas operacionais Linux e Windows. O Linux, por exemplo, utiliza o método l-node, onde são armazenados num disco rígido ou arquivo as informações de permissões de acesso, identificação dos donos dos arquivos, o tamanho do arquivo e os ponteiros. O núcleo do sistema operacional verifica as permissões contidas no l-node do arquivo antes de verificar as permissões dos arquivos ou diretórios. O Sistema de Arquivos do Linux e do Windows consequem trabalhar com uma quantidade grande de arquivos, além de gerenciar arquivos que precisam aumentar o consumo de memória durante a execução. No caso da empresa XYZ Ltda., o mais adequado é gerenciar o espaço em disco por meio do mapa de bits, uma vez que o sistema de arquivos consegue saber rapidamente onde há blocos livres, além de ser simples a sua implementação. O bit igual a 1 corresponde ao bloco que está em uso e o bit 0 ao que não está. Como a empresa XYZ Ltda. é segmentada nas áreas de pré-vendas, vendas, faturamento e administrativo, é necessário garantir a segurança dos diretórios e arquivos de cada setor, para que somente usuários autorizados consigam acessá-los.

Para implementar os mecanismos de proteção, será necessário utilizar os métodos de senha de acesso, grupo de usuários e lista de controle de acesso. No Linux, cada arquivo possui permissão de acesso para ler, gravar e executar, e os tipos de usuários são dono, grupo e outros usuários do sistema. No Windows, a permissão de acesso aos arquivos e pastas pode ser configurada pelo administrador do sistema para cada usuário. O administrador dos sistemas Linux e Windows, na hora da criação dos usuários do sistema, deve atribuir a cada um a permissão (leitura, gravação e execução) ao sistema e diretórios. Durante a criação deles, poderão ser inseridos em um grupo, de acordo com as áreas da empresa (pré-vendas, vendas, faturamento e administrativo), de forma que somente os funcionários de uma determinada área poderão incluir, deletar ou editar os arquivos de seu diretório. Além da senha de acesso e do grupo de usuários, pode ser implementada a lista de controle de acesso (Access Control List - ACL), que é associada a cada arquivo com as permissões de cada usuário. Desta forma, a segurança é reforçada, pois quando um usuário tenta acessar um arquivo, o sistema operacional verifica sua permissão para autorizar ou não a operação realizada.

Para finalizar suas atividades, agora você deverá elaborar o anteprojeto que será entregue para a avaliação dos gestores, apresentando as informações levantadas durante a sua atividade na empresa. Ele deverá conter os seguintes dados:

- O levantamento dos documentos que estavam armazenados dentro de caixas no depósito, analisando a estrutura desses arquivos;
- A definição do sistema de arquivos a ser utilizado nos sistemas operacionais Linux e Windows;
- A definição dos nomes, atributos e operações realizadas, com os arquivos e diretórios que atenderão a cada setor da empresa;
- Os métodos de implantação de arquivos no Linux e Windows
- A implantação do mecanismo de proteção, utilizando os métodos de senha de acesso, grupo de usuários e lista de controle de acesso.

Avançando na prática

Segurança do sistema de arquivos

Descrição da situação-problema

Uma empresa de leilão on-line contratou diversos funcionários para auxiliarem na forca tarefa de vendas de produtos, batizada de Fidelização de Clientes de Produtos on-line. No site da empresa são vendidos produtos de vários tipos e marcas, de lojas virtuais, com descontos de até 70%. A empresa trabalha com máquinas Windows. Você foi contratado como técnico de TI e sua primeira atividade é criar as contas de usuário para cada funcionário admitido para trabalhar na força tarefa. Para evitar qualquer problema no sistema de arquivos, antes da utilização por parte dos novos funcionários, você fez o backup dos arquivos e rodou o antivírus no servidor Windows. No dia seguinte à criação de contas e backup, um coordenador da área de vendas identificou que vários arquivos de fidelização de vendas salvos não estavam mais disponíveis. Ele entrou em contato com a TI, relatando o acontecido. Você foi questionado sobre o desaparecimento dos arquivos, uma vez que o sistema possui segurança e controle de usuários. Como fazer para reverter esta situação?

Resolução da situação-problema

Em algumas situações, pode acontecer de sumirem alguns arquivos do computador, mesmo com a proteção dos arquivos por parte do sistema operacional. Por exemplo, no Windows, em função da criação de novas contas de usuários, por vírus deletados ou até mesmo por alguma falha no disco rígido, pode acontecer a perda de dados pessoais. Como você criou as contas dos usuários e rodou o antivírus do computador, isso pode ter ocasionado o desaparecimento dos arquivos que provavelmente estavam infectados com algum vírus, ou, durante a criação destas contas, o sistema operacional ter criado uma nova conta para o coordenador de vendas. Neste caso, basta acessar o gerenciador de contas de usuários que a conta antiga do coordenador de vendas estará lá com todos os arquivos. Também pode ser feita a recuperação dos arquivos de backup para recompor os arquivos que sumiram.

Faça valer a pena

- **1.** Com relação aos métodos de implementação do sistema de arquivos, analise as questões a seguir e associe as colunas de acordo com a característica de cada método:
- 1 Alocação Contígua
- 2 Alocação Por Lista Encadeada
- 3 Alocação Por Lista Encadeada usando uma tabela na memória
- 4 I-nodes

- () nesse método, os atributos e os endereços em disco dos blocos de arquivos são associados a uma estrutura de dados.
- () nesse método, o acesso aleatório aos arquivos é mais fácil, porém a tabela deve estar na memória o tempo todo.
- () nesse método, a estrutura de dados só precisa estar na memória enquanto o arquivo estiver aberto.
- () nesse método, o acesso aleatório aos arquivos é muito lento.
- () esse método spresenta a vantagem de possuir um bom desempenho e a desvantagem de fragmentar o disco.

Assinale a alternativa que possui a ordem correta da associação das duas colunas:

- a) 4, 2, 4, 1, 2.
- b) 2, 1, 3, 4, 3.
- c) 4, 3, 4, 2, 1.
- d) 1, 3, 4, 2, 4.
- e) 3, 4, 3, 2, 1.
- **2.** Ao criar um arquivo, é necessário gerenciar o espaço livre, identificando quais blocos e setores estão livres para criação dos arquivos. De acordo com os métodos de Gerenciamento de Espaço em Disco, analise as afirmações a seguir.
- I No método mapa de bits, para cada bloco do sistema de arquivos é definido um bit
- II No método da lista encadeada de blocos, cada bloco contém a quantidade de blocos livres que puder ter e possui a localização dos blocos livres.
- III No mapa de bits, o bit igual a 1 corresponde ao bloco está em uso e o bit 0 ao que não estiver.

IV – a desvantagem do mapa de bits em relação à lista encadeada de blocos é que o sistema de arquivos consegue saber rapidamente onde há blocos livres.

Assinale a alternativa correta:

- a) I, II e III estão corretas.
- b) I. III e IV estão corretas.
- c) II. III e IV estão corretas.
- d) l e ll estão corretas.
- e) III e IV estão corretas.

3.



De acordo com uma pesquisa recente da ESET, 87% dos usuários fazem backup de suas informações, principalmente em HDs externos e depois na nuvem, escolhendo prioritariamente documentos de trabalho ou estudo, fotos e senhas. Entre as principais razões, 32% dizem que é por medo de defeito ou erro de hardware, 21% por roubo de informação e 20% por informação corrompida. Por outro lado, entre os usuários que não realizam backup, 72% mencionaram que isso se deve ao esquecimento e, por não saber fazê-lo, 78% perderam informações e 31% perderam dinheiro.

GUTIERREZ, C. **Backup: a importância das cópias de segurança.** 8 abr. 2018. Disponível em: http://www.administradores.com.br/noticias/tecnologia/backup-a-importancia-das-copias-de-seguranca/124318/. Acesso em: 5 jun. 2018.

Com relação aos mecanismos de proteção dos sistemas operacionais, analise as asserções a seguir:

Os sistemas operacionais devem fornecer técnicas para a recuperação dos dados afetados em caso de falhas no sistema de arquivos. A maioria dos sistemas operacionais implementam técnicas de cópias de segurança (backups), de forma que o sistema recupere os dados perdidos em caso de um desastre.

POR QUE

Os dados corrompidos ou perdidos, especialmente dados críticos de negócios coletados em um backup, podem representar a sobrevivência de uma empresa.

A respeito dessas asserções, assinale a opção CORRETA.

- a) As asserções I e II são proposições verdadeiras, e a II é uma justificativa da I.
- b) As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa da I.
- c) A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.
- d) A asserção I é uma proposição falsa, e a II é uma proposição verdadeira.
- e) As asserções I e II são proposições falsas.

Referências

ALENCAR, F. Entenda o que é sistema de arquivos e sua utilidade no PC e no celular. **TechTudo.** 26 fev. 2016. Disponível em: http://www.techtudo.com.br/dicas-etutoriais/noticia/2016/02/entenda-o-que-e-sistema-de-arquivos-e-sua-utilidade-no-pc-e-no-celular.html. Acesso em: 7 maio 2018.

COTA, A. **Diferenças entre o sistema de arquivos do Windows e Linux**. 17 mar. 2005. Disponível em: https://www.vivaolinux.com.br/artigo/Diferencas-entre-o-sistema-de-arquivos-do-Windows-e-Linux>. Acesso em: 11 maio 2018.

COUTO, E. **O que é um inode?** 23 set. 2009. Disponível em: https://gnulinuxbr.wordpress.com/category/comandos-intermediarios/o-que-e-um-inode/. Acesso em: 22 maio 2018.

DEITELL, H. M.; DEITEL, P. J.; CHOFFNES, D. R. **Sistemas Operacionais**. 3. ed. São Paulo: Prentice Hall, 2005.

EASEUS. Meus Arquivos de Computador Sumiram. Preciso de Ajuda para Encontrá-los. 8 jan. 2018. Disponível em: https://br.easeus.com/file-recovery/recuperar-arquivos-e-pastas-sumiram-do-computador-windows-10.html. Acesso em: 24 maio 2018.

GALVIN, P. B., GAGNE, G.; SILBERSCHATZ, A. **Operating system concepts**. 8. ed. United States of America: John Wiley & Sons, Inc., 2013.

GUTIERREZ, C. Backup: a importância das cópias de segurança. 8 abr. 2018. Disponível em: http://www.administradores.com.br/noticias/tecnologia/backup-a-importancia-das-copias-de-seguranca/124318/. Acesso em: 5 jun. 2018.

HAMMERSCHMIDT, R. Microsoft revela mais detalhes sobre o ReFS, seu novo sistema de arquivos. **Tecmundo.** 17 jan. 2012. Disponível em: https://www.tecmundo.com.br/windows-8/17975-microsoft-revela-mais-detalhes-sobre-o-refs-seu-novo-sistema-de-arquivos.htm. Acesso em: 10 maio 2018.

JONES, M. **Anatomia do Sistema de Arquivos do Linux**. 30 out. 2007. Disponível em: https://www.ibm.com/developerworks/br/library/l-linux-filesystem/index.html. Acesso em: 9 maio 2018.

JUNIOR, J. S. V. **Diretórios do Linux**. 10 set. 2009. Disponível em: https://www.vivaolinux.com.br/dica/Diretorios-do-Linux. Acesso em: 15 maio 2018.

JUNIOR, L. A. P. L. **Sistemas de Arquivos**. [S.d.]. Disponível em: http://www.ppgia.pucpr.br/~laplima/ensino/so/materia/04_arquivos.html. Acesso em: 21 maio 2018.

LINUXANDO. **Compreender o Inode em Linux**. [S.d.]. Disponível em: https://www.linuxando.com/tutorial.php?t=Compreender%20o%20Inode%20em%20Linux_7. Acesso em: 22 maio 2018.

MACHADO, F. B.; MAIA, L. P. Arquitetura de Sistemas Operacionais. 4. ed. Rio de Janeiro: LTC. 2007.

- PAULO, D. **Fundamentos do sistema Linux -** arquivos e diretórios. 28 out. 2005. Disponível em: https://www.vivaolinux.com.br/artigo/Fundamentos-do-sistema-Linux-arquivos-e-diretorios/>. Acesso em: 9 maio 2018.
- SANTOS, T. **A Arvore de diretórios do Linux**. 13 out. 2016. Disponível em: https://fsocietybrasil.org/a-arvore-de-diretorios-do-linux/. Acesso em: 11 maio 2018.
- SHIELDS, I. **Gerenciamento de arquivo e diretório**. 10 nov. 2009. Disponível em: https://www.ibm.com/developerworks/br/linux/library/l-lpic1-v3-103-3/index.html. Acesso em: 14 maio 2018.
- SILVA, R. R. **Sistemas de arquivos suportados pelo Linux.** 13 nov. 2006. Disponível em: https://www.vivaolinux.com.br/artigo/Linux-Sistema-de-arquivos>. Acesso em: 10 maio 2018.
- TANENBAUM, A. S. Sistemas Operacionais Modernos. 2. ed. São Paulo: Pearson, 2003.

WALTRICK, R. Saiba como proteger seus arquivos salvos na nuvem. **Gazeta do Povo.** 3 set. 2014. Disponível em: http://www.gazetadopovo.com.br/tecnologia/saiba-como-proteger-seus-arquivos-salvos-na-nuvem-ecz71ltnny743iyd4u3yeas7i. Acesso em: 8 maio 2018.